

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Тернопільський національний технічний університет**  
**імені Івана Пулюя**

**Кафедра автоматизації**  
**технологічних**  
**процесів та виробництв**

**Методичні вказівки**  
**для виконання лабораторної роботи №3 “Математичні**  
**операції (підпрограма множення) на програмному**  
**симуляторі PIC Simulator IDE”**  
**з курсу “Проектування мікропроцесорних систем**  
**керування технологічними процесами”**

**Тернопіль 2017**

Методичні вказівки для виконання лабораторної роботи №3 «Математичні операції (підпрограма множення) на програмному симуляторі PIC Simulator IDE» з курсу «Проектування мікропроцесорних систем керування технологічними процесами».

Методичні вказівки розглянуті і схвалені кафедрою «Автоматизація технологічних процесів та виробництв», протокол № 4 від 21.11.2016 р.

Відповідальні за випуск

доцент, к.т.н. Медвідь В.Р.,  
асистент Пісьціо В.П.

### Лабораторна робота №3

## Математичні операції (підпрограма множення) на програмному симуляторі PIC Simulator IDE

### 1. Робота з програмним симулятором PIC Simulator IDE

Запустивши на виконання PIC Simulator IDE, побачимо основне вікно цієї програми (рис. 1).

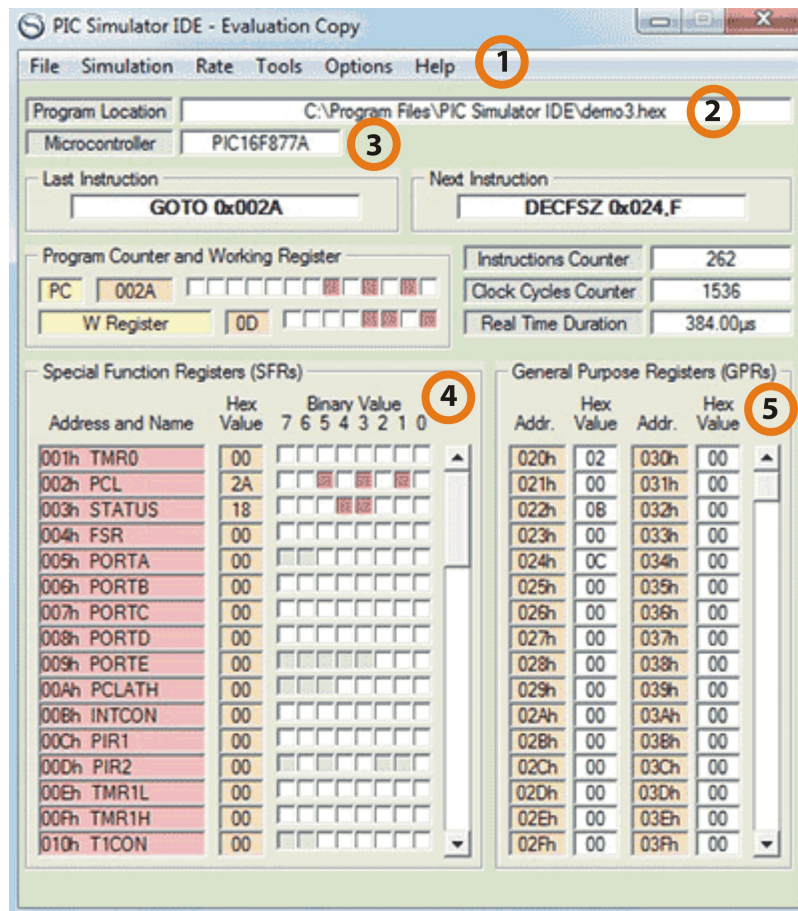


Рис. 1. Основне вікно програми PIC Simulator IDE

У верхній частині знаходяться різні меню, через які можна отримати доступ до різних основних і додаткових модулів програми (на рис. 1 позначено як «1»).

Далі, в рядку Program Location вказано шлях до обраної програми і її ім'я (на рис. 1 - «2»).

Нижче, в рядку Microcontrollers, відображається тип обраного мікроконтролера (на рис. 1 - «3»).

У нижній частині вікна є дві панелі (позначені як «4» і «5»). У них відображаються стан програми, вміст спеціальних і керуючих регістрів обраного МК.

Послідовність роботи з програмним симулятором наступний:

- запуск програми PIC Simulator IDE;
- вибір типу мікроконтролера, для якого написана програма;
- вибір частоти кварцового генератора (впливає тільки на відображувані програмою дані про час виконання програми або команди, але не на швидкість роботи програми, що налагоджуються в PIC Simulator IDE);
- завантаження програми у вигляді HEX-файлу або запуск вбудованого компілятора мови асемблера і написання в ньому потрібної програми;
- вибір потрібних модулів віртуальних пристроїв;

- вибір швидкості і режиму роботи програми симулятора;
- запуск процесу симуляції роботи програми на обраному МК.

Якщо потрібно скористатися для роботи з симулятором власною програмою або внести зміни у вже розроблену, необхідно створити або завантажити для цього файл асемблера, з якого після компіляції буде створений необхідний для роботи з симулятором hex-файл.

Для цього:

1. Натиснути Options | Assembler. Відкриється вікно компілятора Assembler – UNTITLED (рис. 2);

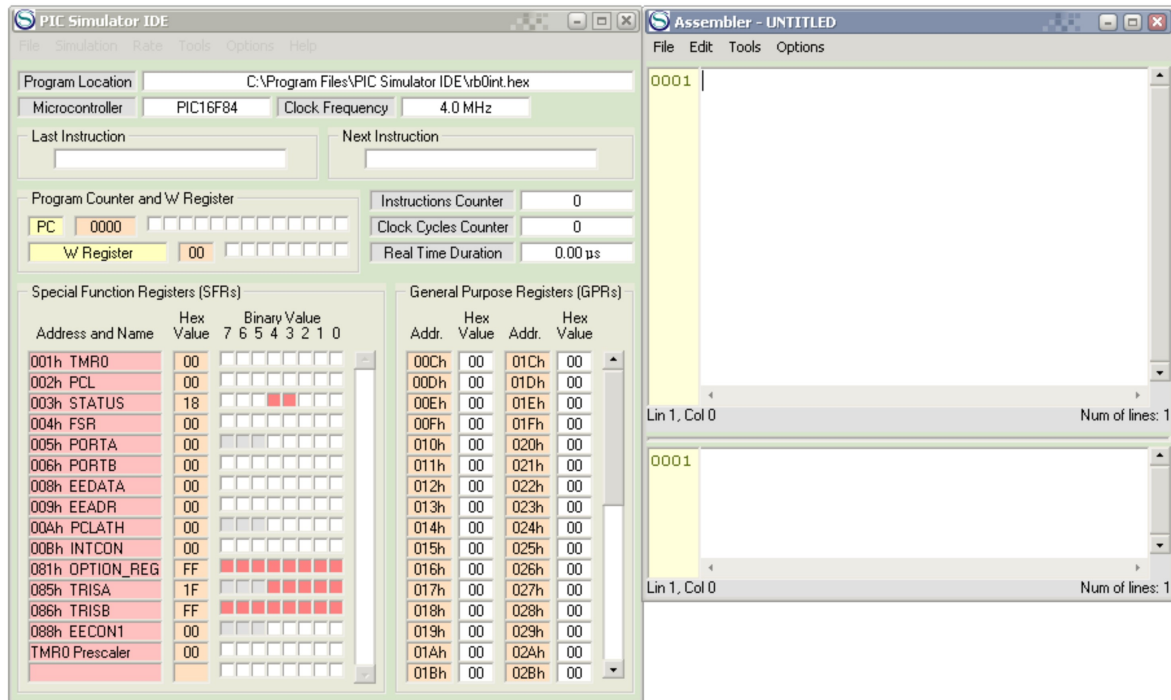


Рис. 2 Вікно симулятора з відкритим вікном Assembler

2. У вікні Assembler натисніть опцію File. Розкриється закладка (рис. 3), з якої для створення нового файлу потрібно натиснути New, а для завантаження вже створеного – OPEN.

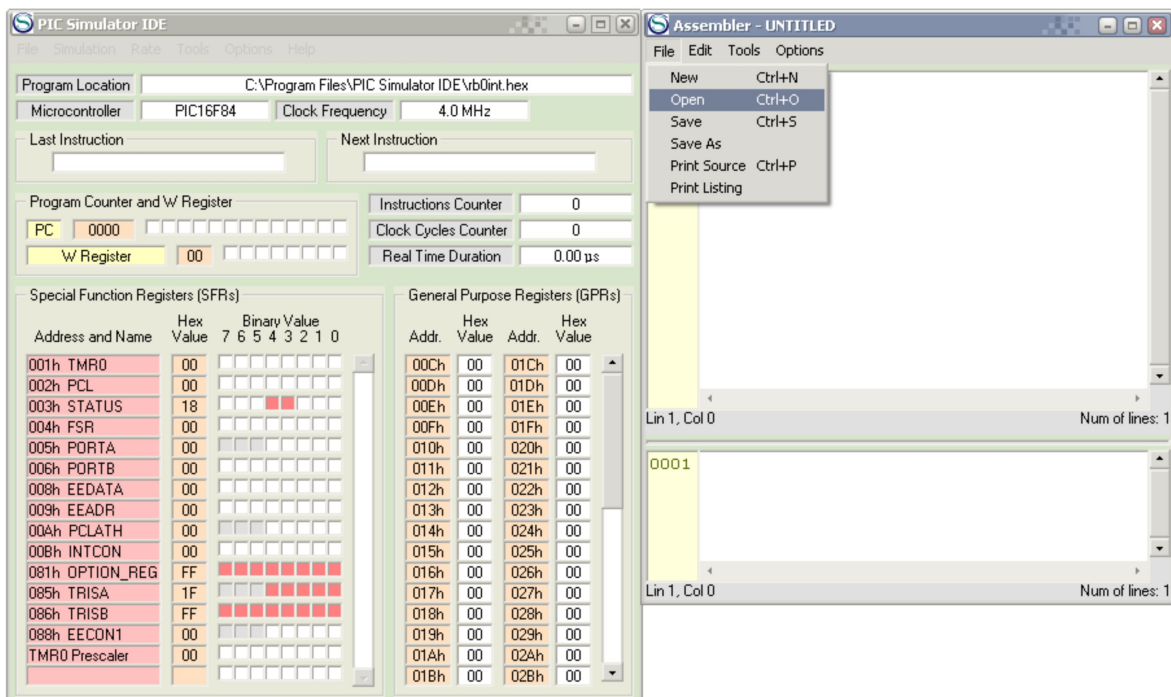


Рис. 3

3. Після вибору і завантаження файлу (наприклад. rb0int.asm), його текст з'явиться в верхній половині вікна Assembler (рис. 4).

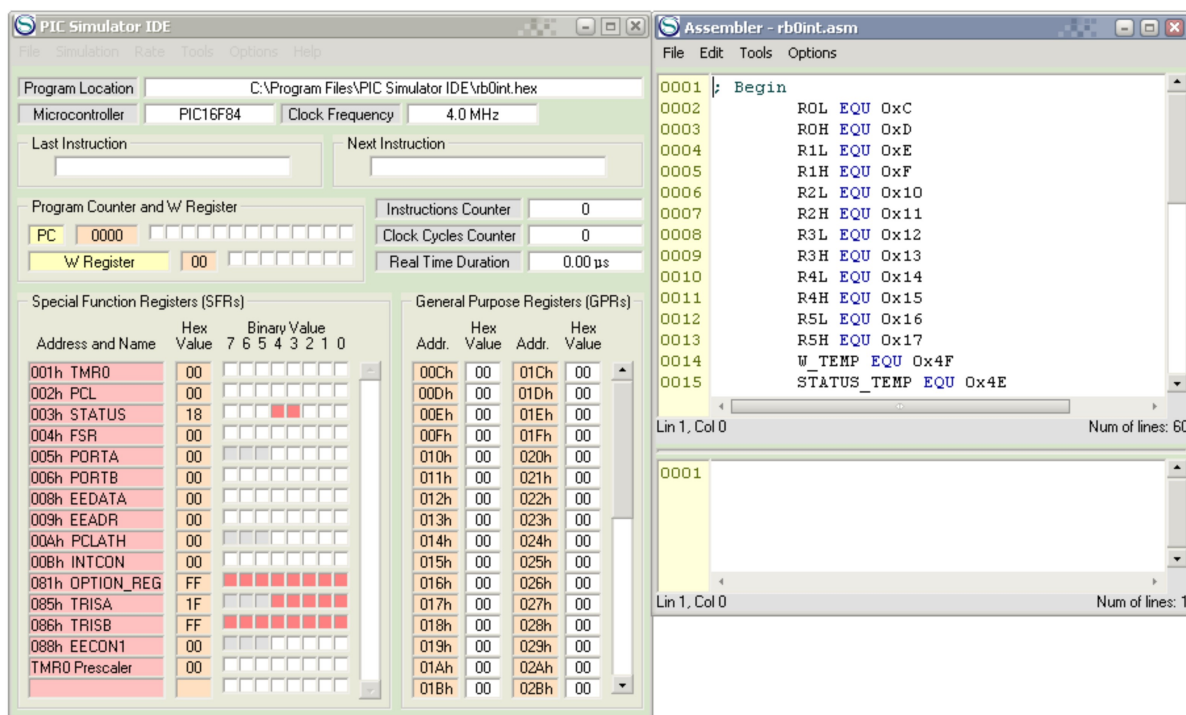


Рис. 4 Завантаження файлу rb0int.asm

4. Для компіляції створеного або завантаженого і потім зміненого файлу, натисніть Tools і у вікні, що розкриється – Assemble. В нижній половині вікна Assembler з'явиться відкомпільований файл і одночасно, при відсутності помилок, буде створений одноіменний hex-файл.

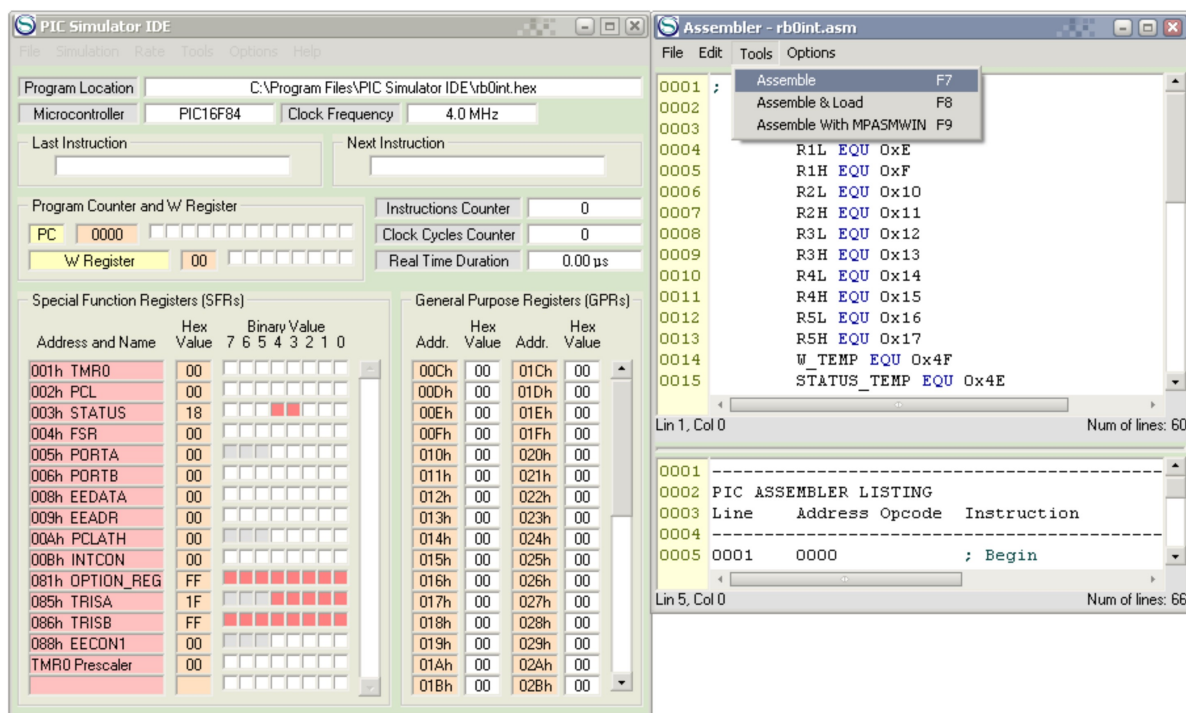


Рис. 5

**2. Завдання на лабораторну роботу:** математичні операції: підпрограма множення..

1. Вивчити програмну модель PIC Simulator IDE.

2. Вивчити команди арифметичних операцій PIC – контролера.

3. Написати і дослідити роботу програми з Прикладу 1 та дослідити вміст регістрів контролера (W, STATUS), які використовуються при виконанні цієї програми.

4. Користуючись вікном “PIC Disassembler”, записати перші 7 команд виконуваної програми на Асемблері та в шістнадцяткових кодах.

5. Записати для вибраних команд коментар щодо їх призначення (див. Приклад 2).

### Приклад 1.

Демонстрація роботи симулятора при виконанні арифметичних операцій. У цьому прикладі перемножуються два числа: 123 (шістнадцятковий 7B) і 234 (шістнадцятковий EA), отримаємо результат 28782 (шістнадцятковий 706E).

Текст програми з файлу multiply.asm має наступний вигляд:

; Begin

```
R0L EQU 0xC
R0H EQU 0xD
R1L EQU 0xE
R1H EQU 0xF
R2L EQU 0x10
R2H EQU 0x11
R3L EQU 0x12
R3H EQU 0x13
R4L EQU 0x14
R4H EQU 0x15
R5L EQU 0x16
R5H EQU 0x17
ORG 0x0000
BCF PCLATH,3
BCF PCLATH,4
GOTO L0001
ORG 0x0004
RETFIE
```

L0001:

; 1: Dim a As Word 'first number

; The address of 'a' is 0x18

a EQU 0x18

; 2: Dim b As Word 'second number

; The address of 'b' is 0x1A

b EQU 0x1A

; 3: Dim x As Word 'result

; The address of 'x' is 0x1C

x EQU 0x1C

; 4:

; 5: a = 123 'set first number

MOVLW 0x7B

MOVWF 0x18

CLRF 0x19

; 6: b = 234 'set second number

MOVLW 0xEA

MOVWF 0x1A



```

        CLRF 0x1B
; 7: x = a * b 'calculate result
        MOVF 0x18,W
        MOVWF R3L
        MOVF 0x19,W
        MOVWF R3H
        MOVF 0x1A,W
        MOVWF R1L
        MOVF 0x1B,W
        MOVWF R1H
        CALL M001
        MOVWF 0x1C
        MOVF R2H,W
        MOVWF 0x1D
; End of program
L0002:GOTO L0002
; Multiplication Routine
M001: MOVLW 0x10
        MOVWF R4L
        CLRF R0H
        CLRF R0L
M002: RRF R3H,F
        RRF R3L,F
        BTFSS STATUS,C
        GOTO M003
        MOVF R1L,W
        ADDWF R0L,F
        MOVF R1H,W
        BTFSC STATUS,C
        INCFSZ R1H,W
        ADDWF R0H,F
M003: RRF R0H,F
        RRF R0L,F
        RRF R2H,F
        RRF R2L,F
        DECFSZ R4L,F
        GOTO M002
        MOVF R2L,W
        RETURN
; End of listing
END

```

### **3. Послідовність роботи з симулятором при виконанні програми**

Запустимо цю програму в PIC Simulator IDE і виконаємо наступне:

1. Запустити PIC Simulator IDE;
2. Натиснути Options | Select Microcontroller;
3. Вибрати PIC16F84 і натиснути кнопку Select;
4. Натиснути File | Load Program;
5. Вибрати файл multiply.hex і натиснути кнопку Open;
6. Натиснути Tools | Breakpoints Manager (відкриється вікно Breakpoints Manager);
7. Натиснути «Так», щоб використовувати існуючий асемблер, який видав файл;

8. Натиснути рядок, що відповідає адресі 0018, щоб встановити контрольну точку на цій команді;
9. Вибрати Hold PC In Focus option;
10. Вибрати Rate | Extremely Fast simulation rate;
11. Натиснути Simulation | Start (почнеться моделювання).

Коли ця математична підпрограма закінчиться, програма входить в нескінченний цикл за адресою 0018, але через контрольні точки PIC Simulator IDE автоматично переключиться в режим моделювання Step by step.

Роботу PIC Simulator IDE можна зупинити, натиснувши на Simulation | Stop, або продовжити її виконання, очистивши контрольну точку і натиснувши на Rate | Extremely.

Регістри 19 та 18 будуть містити перший параметр: 007B.

Регістри 1BH і 1AH міститимуть другий параметр: 00EA. Результат 706E знаходиться в регістрах 1DH і 1CH.

Вигляд екрану з виконуваною програмою показаний на рис. 6.

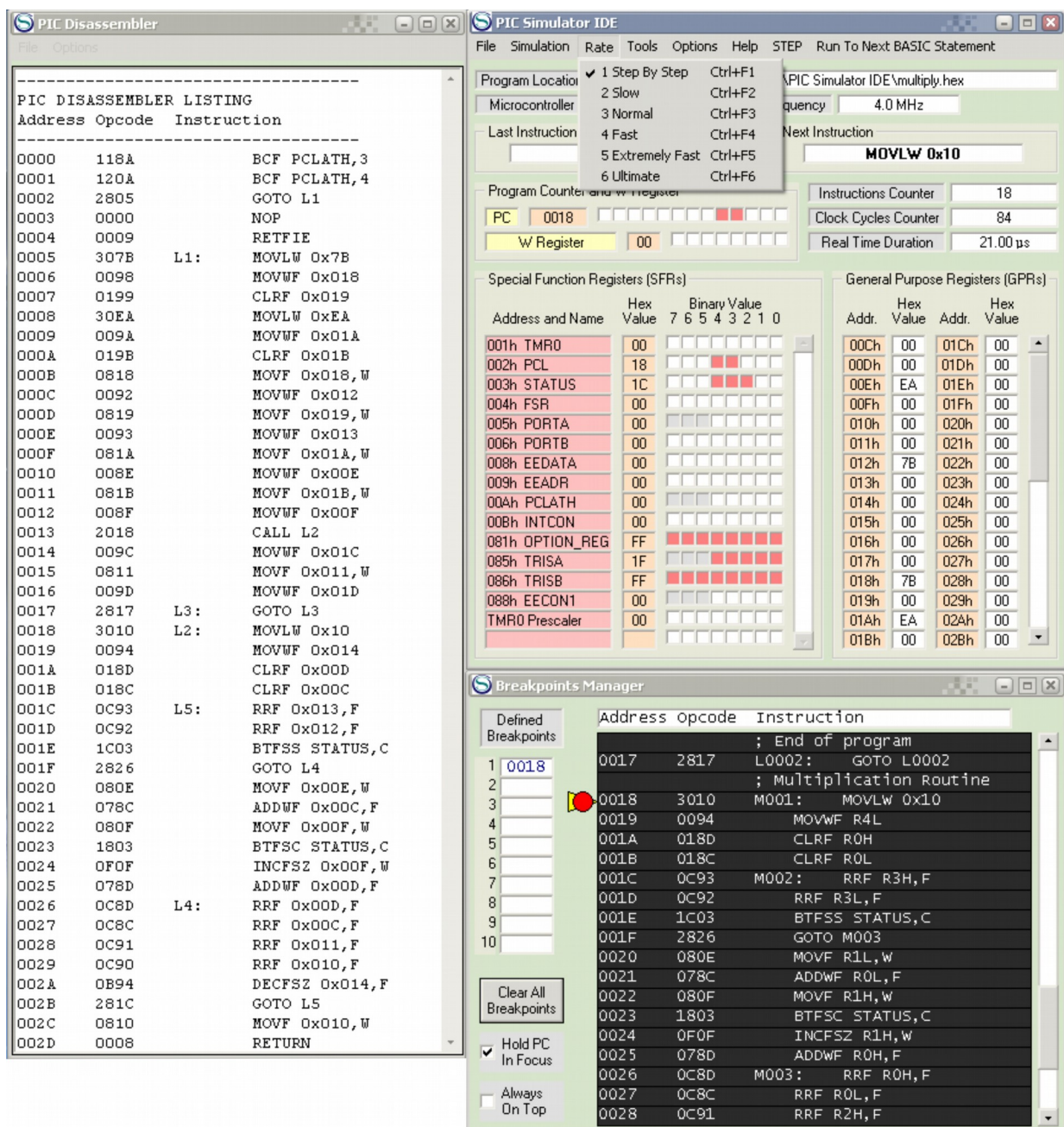


Рис. 6 Вигляд екрану з програмою «Математичні операції: підпрограма множення»



З вікна “PIC Disassembler” вибираємо сім перших команд з їх шістнадцятковими кодами і знаходимо з таблиці кодів асемблера PIC-контролера коментар щодо призначення цих команд ( див. Приклад 2, де наведено такий запис для однієї команди).

### Приклад 2

#### Код команди

118A

#### Команда

BCF PCLATH, 3

#### Виконувана операція (коментар)

; скинути в “0” 3-ій біт регістра PCLATH

і т.д.

Вміст регістрів контролера, які використовуються при виконанні програми, знаходимо з області регістрів Address and Name, яка розташована в лівій нижній частині основного вікна симулятора (виділені рожевим кольором). Всі регістри вольовозрядні.

В процесі виконання програми по зміні кольору комірок видно, вміст яких регістрів змінюється. Забарвлення комірки відповідного розряду регістру помаранчевим кольором означає наявність “1”, білим - “0”. Вміст регістрів записуємо в шістнадцятковому коді за Прикладом 3.

### Приклад 3

#### Регістр

PORTA

#### Вміст регістра

1F

і т. д.

## 4. Контрольні запитання

1. Призначення регістра W мікроконтролера.
2. Арифметичні команди мікроконтролера.
3. Формат регістра STATUS.
4. Призначення та позначення основних елементів програмної моделі мікроконтролера

## 5. Література

1. Данилин А. Программа-симулятор PIC Simulator IDE / Данилин А. // Современная электроника. 2006.- №4. -С. 68-76.
2. Тавернье К. PIC-микроконтроллеры. Практика применения. М.: ДМК, 2002.
3. Предко М. Создайте работа своими руками на PIC- контроллере./ Майкл Предко; Пер. с английского Земского Ю.В. – М.: ДМК Пресс, 2006. – 408 с.: ил. – (В помощь радиолюбителю).
4. Кениг А. и М. Полное руководство по PIC-микроконтроллерам.: Пер. с нем.-К.: “МК-Пресс”, 2007.-256 с., ил.